REMARKS

Claims 10, 11, 14 and 15 remain pending, the independent claims 10 and 14 having been amended to more specifically define the invention. New dependent claim 16 is being added.

Dependent claim 15, objected to because of use of the acronym "FAT" has been amended to include the descriptive term "file allocation table."

Claims 10, 11, 14 and 15 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. patent application publication no. 2003/0028704A1 to Mukaida et al. (hereinafter "Mukaida") in view of U.S. patent application publication no. 2004/0030825A1 to Otake et al. (hereinafter "Otake").

Claims 10 and 11

It is respectfully submitted that neither of the cited Mukaida nor Otake references, either alone or in combination, disclose executing write commands received by the flash memory system by either storing the data in parallel in multiple pages of a metablock extending across multiple memory planes or sequentially in blocks of a single plane, depending upon the amount data received with the write commands. Specifically, it is submitted that neither reference discloses writing data received with a write command "sequentially into pages within individual blocks of *only one* of the planes" (claim 10, emphasis added), in combination with also writing data received with other write commands into metablocks formed across two or more planes.

The Office Action (p. 3, lines 13-16) alleges that Mukaida discloses "variously writing the received data in parallel sequentially into pages within individual blocks of one of the planes (figure 12, and pages 10-11 [0180] to [0193]) in response to characteristics of the host write command (figure 22, page 17 [0289] to [0297], and pages 20-20, [0345] to [0346])." It is not agreed that this anticipates the single page write limitation of claim 10.

Mukaida's figure 12 shows executing a write command by writing data into each of flash memory chips 2-0 to 2-3 one at a time, and the cited text describes this. The data are not written into individual blocks of *only one* of the planes as specified in claim 10. The data are clearly shown to be written in all four of Mukaida's flash memory chips, although sequentially in one chip at a time. This writing scheme is not limited to writing the data in sequence in only one of the planes.

Mukaida's figure 22, and cited text describing it, is even further removed from the single plane limitation of claim 10 because it writes data into all four banks #0 – #3 in parallel. Data for each of the banks are loaded into the memory chip in time sequence but once loaded are then flash programmed simultaneously into all the four banks.

The cited paragraphs 0345-0346 of page 20 of Mukaida describe yet a different programming technique that is illustrated in its figure 29. Multiple banks of one flash memory chip are flashed programmed simultaneously, followed by multiple banks of another flash memory chip. This is also submitted to have nothing to do with the claimed alternative of writing data sequentially in only one of the memory planes.

The present application (paragraph 0013) best summarizes the benefit of the claimed technique wherein data are written either into a metablock across multiple memory system planes or in only one of the planes, as follows:

> Sectors of data are written in parallel to all blocks of a metablock, when the number of data sectors being programmed relative to the capacity of the metablock make the advantages of a high degree of parallelism worthwhile. However, when a particular data programming operation involves only a small number of data sectors relative to the storage capacity of the metablock, resulting increased data consolidation operations can hurt performance more than the higher parallelism will help, in which case the advantages of parallelism provided by the use of metablocks are not realized.

The technique is particularly useful when the memory system is being used with a host that frequently issues commands to write one sector or one page of data per command. The advantages of a maximum parallel write of single sectors or pages in this case can be overcome by the disadvantages of the resulting increased data consolidation (garbage collection) that the single sector or page writes will cause. With the high degree of parallelism, unrelated data sectors or pages are programmed together but yet they are independently updated or replaced. This causes the original sectors or pages to be rendered obsolete, which then requires consolidation of remaining valid data in order to reclaim the memory capacity occupied by the obsolete data.

Therefore, for these reasons, it is respectfully submitted that claim 10, and thus also its dependent claim 11, are patentable over the cited Mukaida and Otake references. Even if their

disclosures are somehow fit together, such a combination will not include the single plane programming as specified in amended claim 10.

Claims 14 and 15

Independent claim 14 is believed to be patentable for the same reasons given above for claim 10. Claim 14 specifies the storage of data either in parallel in more than one sub-array or in a single sub-array, depending upon whether multiple pages or only one page of data are received with the write command. This is recited in the two paragraphs of claim 14 that each begin with "in response to receiving the write commands..."

Dependent claim 15 details one of the applications of the method of claim 14 that is quite important. Updating the FAT each time new data are written, or when existing data are updated, typically results in many repetitive single sector or page writes to the memory. This adversely affects the performance of the memory system but the technique of claim 14 overcomes this by treating single page writes differently from multiple page writes. Nothing in either of the cited Mukaida or Otake references has been found to suggest this advantageous single page writes of FAT data.

In addition, the last paragraph of claim 14 calls for "maintaining indications" associated with the individual written data sectors that indicate whether or not the sector was written into the memory in logical sequence with other sectors. The Office Action (p. 6, lines 7-19) alleges that this is disclosed by Otake, referencing its paragraphs 0059-0064 that describe the block management table of Figure 3. However, that table provides an entry for each "logical address block" and not for the individual sectors. That table also does not specify that a given sector was written in a single block, as claimed. Further, that table is not "associated with the written sectors of data" as claimed, and certainly not included in headers of the data sectors as recited in new claim 16.

For these reasons, it is respectfully submitted that claim 14, and thus also its dependent claims 15 and 16, are patentable over the cited Mukaida and Otake references. Even if their disclosures are fit together in some manner, such a combination will not include the single plane programming of single pages of data received with individual write commands, as is specified in amended claim 14. Any combination of Mukaida and Otake would also lack disclosure of

maintaining indications of the logical connection of individual sectors with others, as also included in claim 14.

Information Disclosure Statement

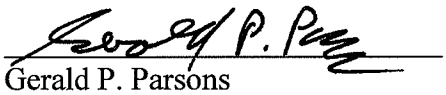
A Supplemental Information Disclosure Statement being filed herewith includes United States publication no. 2005/0144358A1 of another application filed by two of the applicants herein on the same day as the present application. The disclosures of these two applications are different but they are both directed to flash memory operation. The most recent Office Action and responsive Amendment in that other application no. 10/749,831 are also included.

Conclusion

Accordingly, it is believed that this application is now in condition for allowance and an early indication of its allowance is solicited. However, if the Examiner has any further matters that need to be resolved, a telephone call to the undersigned attorney at 415-276-6534 would be appreciated.

FILED VIA EFS

Respectfully submitted,



October 29, 2007

Gerald P. Parsons
Reg. No. 24,486

Date

Davis Wright Tremaine LLP
505 Montgomery Street, Suite 800
San Francisco, CA 94111-6533
(415) 276-6500 (main)
(415) 276-6534 (direct)
(415) 276-6599 (fax)
Email: geraldparsons@dwt.com